


System of Systems Architecture Evaluation with Concurrent Development

SATURN 2007
Mike Gagliardi, Bill Wood

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

 **Software Engineering Institute** | CarnegieMellon

© 2007 Carnegie Mellon University

Problem

Severe integration and operational problems can arise due to inconsistencies in addressing system quality attributes between system and software architecture.

This is further exacerbated in a System of Systems (SoS) context where major system and software elements are developed concurrently and independently by a variety of contractors.

Need a methodology to identify these inconsistencies within a system of systems context early in the life cycle.

Report Documentation Page				Form Approved OMB No. 0704-0188	
Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
1. REPORT DATE MAY 2007		2. REPORT TYPE		3. DATES COVERED 00-00-2007 to 00-00-2007	
4. TITLE AND SUBTITLE System of Systems Architecture Evaluation with Concurrent Development				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Carnegie Mellon University ,Software Engineering Institute (SEI),Pittsburgh,PA,15213				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited					
13. SUPPLEMENTARY NOTES presented at the SEI Software Architecture Technology User Network (SATURN) Workshop, 14-16 May 2007, Pittsburgh, PA.					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT Same as Report (SAR)	18. NUMBER OF PAGES 8	19a. NAME OF RESPONSIBLE PERSON
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified			

Our Intentions

Develop a method based on but not restricted by ATAM and QAW

- Quality attributes, augmented mission threads, and scenarios
- Multi-layered evaluation
 - first pass looking for inconsistencies, also identifying potential systems for further, detailed evaluation (using S-ATAM or ATAM) techniques.
 - Subsequent evaluations of individual systems use mission thread fragments, scenarios from first pass evaluation. May need further decomposition of mission threads, scenarios, etc for “SoS to System” transition.

Work with large-scale programs and contractors to develop techniques and participate in SoS evaluations.



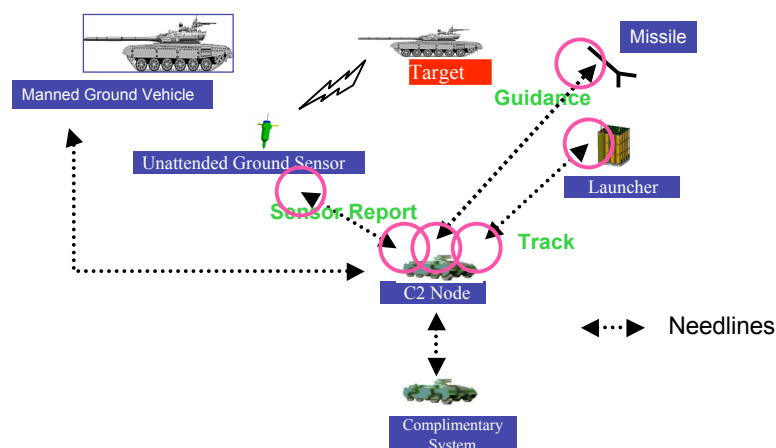
Software Engineering Institute

CarnegieMellon

Gagliardi, Wood
© 2007 Carnegie Mellon University

3

SoS Architecture View (OV-2)



Software Engineering Institute

CarnegieMellon

Gagliardi, Wood
© 2007 Carnegie Mellon University

4



Specific Mission Thread

An enemy tank platoon is threatening a lightly protected company, and comes into the field of view of a UGS, which connects to and informs a MGVC2. An MGVC2 identifies the enemy tanks. The MGVC2 assigns an unmanned missile launcher (UML) to engage the tank platoon. The UML engages and destroys the enemy. The MGVC2 determines that the threat has been eliminated from subsequent UGS signals.



Software Engineering Institute

CarnegieMellon

Gagliardi, Wood
© 2007 Carnegie Mellon University

5

Quality Attribute Augmentations

The engagement will be started within 10 seconds of sensing the enemy tank platoon

Any one failure does not impact any timelines

- Hardware, software, user, vehicle

This happens in an environment where there are 2000 BSEs being tracked

The security level between the MGVC2 and the UML is top secret

Include some measure of (importance, difficulty)



Software Engineering Institute

CarnegieMellon

Gagliardi, Wood
© 2007 Carnegie Mellon University

6



Evaluation

Evaluate high level SoS representations and guidelines

- Cross system mission threads augmentation with quality attributes and high level architecture from different systems
- Only evaluate the areas where the systems interact
- Follow general ATAM steps with key stakeholders; architect walking through architecture using augmented mission threads; evaluation team probing architecture.
- Define gaps and risks and use them to target "focused evaluations" or ATAMs using the specific augmented mission threads for further evaluation
- *Assumes the documentation of SoS architecture guidelines and representations*
- *Use "Mission Thread Augmentation Workshop" early in life cycle*

Evaluate individual systems (or combinations)

- Derive scenarios from the augmented mission threads
- Use "focused evaluations" or ATAMs



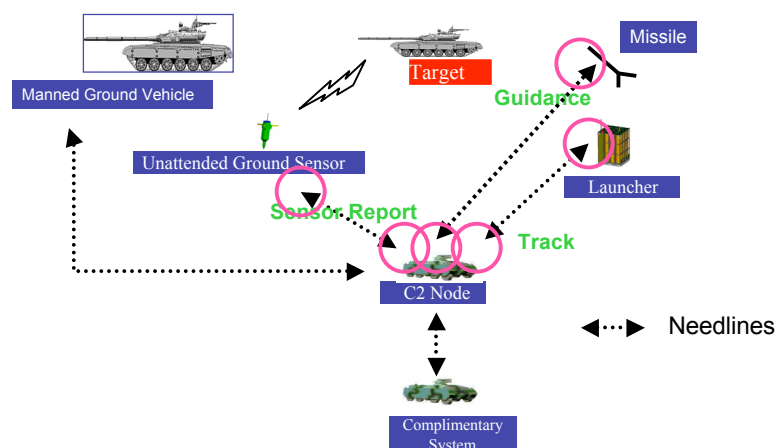
Software Engineering Institute

CarnegieMellon

Gagliardi, Wood
© 2007 Carnegie Mellon University

7

SoS Architecture View (OV-2)



Software Engineering Institute

CarnegieMellon

Gagliardi, Wood
© 2007 Carnegie Mellon University

8



Software Engineering Institute

CarnegieMellon

© 2007 Carnegie Mellon University

4

Concurrent Development

SoS Architecture Guidelines are concurrently developed with some architecture components (views)

- Some folks “get ahead of the guidelines”
- Some legacy components may not be re-engineered to guidelines

Systems are architected and designed concurrently and independently, often with different views and tools. Must reconcile different architecture products as they become available.



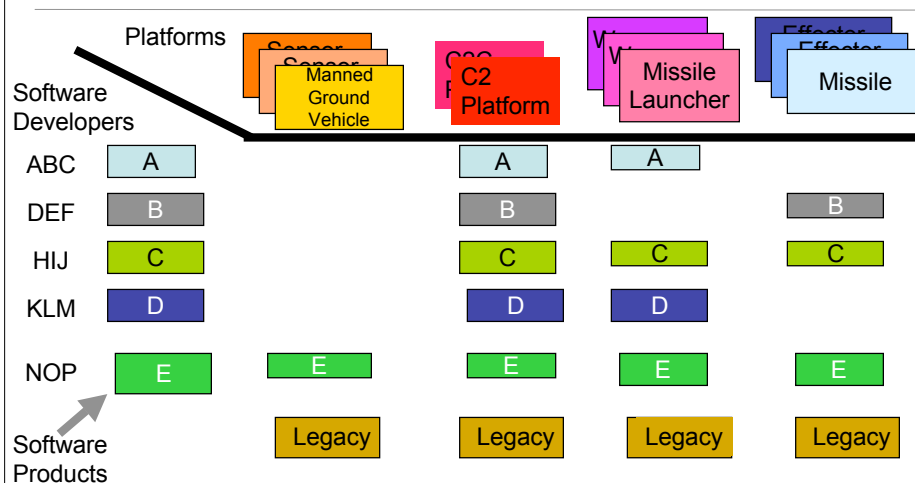
Software Engineering Institute

CarnegieMellon

Gagliardi, Wood
© 2007 Carnegie Mellon University

9

Example: Multiple Contractors



Software Engineering Institute

CarnegieMellon

Gagliardi, Wood
© 2007 Carnegie Mellon University

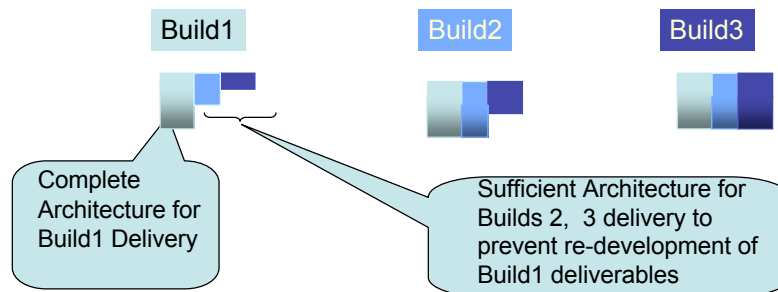
10



Life-Cycle – Multiple Builds

Evaluation should focus on “Next Build”

- Including enough architecture of future build to minimize extensive re-work between builds (especially safety, security, fault tolerance)



Software Engineering Institute

CarnegieMellon

Gagliardi, Wood
© 2007 Carnegie Mellon University

11

Quality Attribute Augmentations (with builds)

The engagement will be started within 10 seconds of sensing the enemy tank platoon **Phase 1**

Any one failure does not impact any timelines

- Hardware, software, user, vehicle – **Phase 2**

This happens in an environment where there are 2000 BSEs being tracked
(1000 in **phase 1**)

The security level between the MGVC2 and the UML is top secret - **phase 3**

Include some measure of (importance, difficulty)



Software Engineering Institute

CarnegieMellon

Gagliardi, Wood
© 2007 Carnegie Mellon University

12



Evaluation (with builds)

Evaluate high level SoS representations and guidelines

- Cross system mission threads augmentation with quality attributes and high level architecture from different systems
- Only evaluate the areas where the systems interact
- Follow general ATAM steps with key stakeholders; architect walking through architecture using augmented mission threads; evaluation team probing architecture.
- Define gaps and risks and use them to target "focused evaluations" or ATAMs using the specific augmented mission threads for further evaluation
- Evaluate each build
- Evaluation focuses on current and next build
 - Including enough architecture of future build to minimize extensive re-work between builds (especially safety, security, fault tolerance)

Evaluate individual systems (or combinations)

- Derive scenarios from the augmented mission threads
- Use "focused evaluations" or ATAMs



Software Engineering Institute

CarnegieMellon

Gagliardi, Wood
© 2007 Carnegie Mellon University

13

Challenges

Technical

- SOA, Enterprise Architecture, legacy / COTS / GOTS components, Massive Middleware products

Evaluation

- Each contractor may use diverse DODAF products, software views and architecture toolset
- Mission threads cross contractor/system/build boundaries, sparse on quality attributes
- SoS requirements often too vague
- Black box nature of many components
- Architecture/Design documentation does not match implementation

Programmatic

- Multiple Contractors' architectural products (views) for systems/software to be integrated into many weapons/sensor platforms and SILs
- Concurrent Development of system and software architectures – tension between system and software architects
- Life-cycle of many builds with spin-outs to the field from each build



Software Engineering Institute

CarnegieMellon

Gagliardi, Wood
© 2007 Carnegie Mellon University

14



